



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/625,908	07/24/2003	Axel Von Bergen	13909-119001 / 2003P00132	1208
32864	7590	12/16/2005	EXAMINER	
FISH & RICHARDSON, P.C. PO BOX 1022 MINNEAPOLIS, MN 55440-1022			BRADLEY, MATTHEW A	
			ART UNIT	PAPER NUMBER
			2187	

DATE MAILED: 12/16/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/625,908	BERGEN ET AL.	
	Examiner	Art Unit	
	Matthew Bradley	2187	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 24 July 2003.
 2a) This action is FINAL. 2b) This action is non-final.
 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-24 is/are pending in the application.
 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
 5) Claim(s) _____ is/are allowed.
 6) Claim(s) 1-24 is/are rejected.
 7) Claim(s) _____ is/are objected to.
 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
 10) The drawing(s) filed on 24 July 2003 is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ . |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>5/17/04, 2/14/05</u> . | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| | 6) <input type="checkbox"/> Other: _____ . |

DETAILED ACTION

Information Disclosure Statement

The information disclosure statement (IDS) submitted on 14 February 2005 was filed after the filing date of 24 July 2003 for application 10/625,908. The submission is in compliance with the provisions of 37 CFR 1.97. Accordingly, the Examiner is considering the information disclosure statement with a signed and initialed copy being attached hereto.

The information disclosure statement (IDS) submitted on 17 May 2004 was filed after the filing date of 24 July 2003 for application 10/625,908. The submission is in compliance with the provisions of 37 CFR 1.97. Accordingly, the Examiner is considering the information disclosure statement with a signed and initialed copy being attached hereto.

Claim Status

Claims 1-24 remain pending and are ready for examination.

Double Patenting

The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA

Art Unit: 2187

1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

Claim 1 is provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claim 11 of copending Application No. 10/625,897.

This is a provisional obviousness-type double patenting rejection because claim 1 of the instant application is anticipated by co-pending Application No. 10/625,897 claim 11 in that claim 11 of the co-pending application contains all the limitations of claim 1 of the instant application. Claim 1 of the instant application therefore is not patently distinct from the co-pending application claim and as such is unpatentable for obvious-type double patenting, as shown in the table below.

Instant Application: 10/625,908 Claim ----- (Limitation)	Co-pending Application 10/625,897 Claim ----- (Limitation)
1 – (a) A method for allocating memory in a computer system, the method comprising:	11 – (a) A method for allocating memory in a computer system, the method comprising:
1 – (b) outputting a request from an application to an operating system for allocation of a block of memory by the	11 – (b) outputting a request from an application to an operating system for allocation of a block of memory by the

Instant Application: 10/625,908 Claim ----- (Limitation)	Co-pending Application 10/625,897 Claim ----- (Limitation)
operating system to the application	operating system to the application;
1 – (c) accessing the block of memory at the application;	11 – (c) accessing the block of memory at the application
1 – (d) dividing the block of memory into frames;	11 – (d) dividing the block of memory into frames;
1 – (e) dividing each of the frames into instances; and	11 – (e) dividing each of the frames into instances
1 – (f) associating an application-defined instance type with the instances for data storage using the instances.	11 – (f) with each instance operable to store data and associated with an application-defined instance type; and
	11 – (g) maintaining a data structure indicating each unused instance.

A later claim that is not patentably distinct from an earlier claim in a commonly owned patent is invalid for obvious type double patenting. *In re Berg*, 140 F.3d 1428, 1431, 46 USPQ2d 1226, 1229 (Fed. Cir. 1998). A later patent claim is not patentably distinct from an earlier patent claim if the later claim is obvious over, or anticipated by, the earlier claim. *In re L2Dgi*, 759 F.2d at 896, 225 USPQ at 651 (affirming a holding of obviousness type double patenting because the claims at issue were obvious over claims in four prior art patents); *In re Berg*, 140 F.3d at 1437, 46 USPQ2d at 1233 (Fed. Cir. 1998) (affirming a holding of obviousness type double patenting where a patent application claim to a genus is anticipated by a patent claim to a species within that genus). " *ELI LILLY AND COMPANY v BARR LABORATORIES, INC.*, United States Court of Appeals for the Federal Circuit, ON PETITION FOR REHEARING EN BANC (DECIDED: May 30, 2001).

"The only other difference between claim 1 of the '213 patent and claim 7 of the '549 patent is that the former is directed to humans while the latter is directed to

animals. Humans are a species of the animal genus. Our case law firmly establishes that a later genus claim limitation is anticipated by, and therefore not patentably distinct from, an earlier species claim. In re Berg, 140 F.3d at 1437, 46 USPQ2d at 1233 (Fed. Cir. 1998); In re Goodman, 11 F.3d 1046, 1053, 29 USPQ2d 2010, 2016(Fed. Cir. 1993); In re Gosteli, 872 F.2d 1008, 1010, 10 USPQ2d 1614, 1616(Fed. Cir. 1989); Titanium Metals Corp. v. Banner, 778 F.2d 775, 782, 227 USPQ 773, 779 (Fed. Cir. 1985); In re Van Ornum, 686 F. 2d at 944, 214 USPQ at 767 (C.C.P.A. 1982)." ELI LILLY AND COMPANY v BARR LABORATORIES, INC., United States Court of Appeals for the Federal Circuit, ON PETITION FOR REHEARING EN BANC (DECIDED: May 30, 2001).

Claim Objections

Claim 1 is objected to because of the following informalities:

The Examiner notes that the first limitation found in the instant claim, claims a request for allocation of memory to a requesting application. The second limitation claims, "accessing the block of memory at the application." The application is previously requesting the allocation of memory yet this limitation is implying that memory is already allocated in the form of a block at the application. The Examiner is unsure if the application is requesting additional memory, but for purpose of examination, the Examiner is interpreting the accessing of the block of memory **for** the application instead of **at** the application.

Any claim not specifically addressed is objected to by virtue of its dependency.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 11-20 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. The Examiner notes that as written, claims 11-20 are drawn to a non-statutory software application. The following amendment to said claims would overcome the instant rejection:

A software application tangibly embodied on a computer readable medium using application-level memory management, the software application comprising:

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-24 are rejected under 35 U.S.C. 102(b) as being anticipated by McMahon et al. (U.S. 5,784,699) herein after referred to as McMahon.

As per independent claim 1, McMahon teach,

- outputting a request from an application to an operating system for allocation of a block of memory by the operating system to the application;
(Column 5 lines 25-27)
- accessing the block of memory at the application; (Column 5 lines 30-39)

- o dividing the block of memory into frames; (Column 4 lines 37-40)
- o dividing each of the frames into instances; and associating an application-defined instance type with the instances for data storage using the instances. (Column 5 lines 50-59).
 - o *The Examiner notes that the dynamic memory allocator takes the blocks (frames) from the whole memory and subsequently divides the blocks (frames) into portions (instances) for allocations. As the request for memory from the application, comes from the application, any allocated memory for that application will contain space available for that application to use. Accordingly, the allocation of memory for the application will contain data storage for the instances using the instances of the requesting application.*

As per dependent claim 2, McMahon teach, allocating a block of memory that begins on a page boundary (Column 3 lines 28-37). *The Examiner notes that a free list is present in the system of McMahon. As shown in the table in Column 6 of McMahon, the dynamic memory allocator, in the form of these free lists, catalogs various free blocks of memory by size. Accordingly, when an allocation takes place by the dynamic memory allocator, for example of the 16-byte size, the memory allocated will begin on the 16-byte boundary.*

As per dependent claim 3, McMahon teach, wherein the size of the block of memory is determined by a coding parameter associated with the application (Column 3 lines 6-8).

As per dependent claim 4, McMahon teach, wherein dividing the block of memory into frames includes identifying a first page boundary within the block of memory (Column 3 lines 28-37). *The Examiner notes that a free list is present in the system of McMahon. As shown in the table in Column 6 of McMahon, the dynamic memory allocator, in the form of these free lists, catalogs various free blocks of memory by size thereby indicating a boundary for each size that is maintained.*

As per dependent claim 5, McMahon teach, wherein dividing the block of memory into frames further includes designating a portion of the block of memory before the first page boundary as unused (Column 6 lines 34-37). *The Examiner notes that as discussed supra and with respect to the instant citation, the allocator is able to divide the memory block and release the remainder of the block as free space. Accordingly, the allocator designates a portion of the block of memory as unused.*

As per dependent claim 6, McMahon teach, wherein a size of each frame is determined by a coding parameter (Column 3 lines 6-8).

As per dependent claim 7, McMahon teach, wherein a size of each frame is determined by a page size used by the operating system (Column 5 lines 8-21). *The Examiner notes that the allocator allocates virtual pages that originate from the operating system. Accordingly, the size of each frame is determined by the operating system's page size upon allocation.*

As per dependent claim 8, McMahon teach, wherein dividing a block of memory into frames includes: determining a last page boundary within the block of memory; and designating a portion of the block of memory after the last page boundary as unused

(Column 3 lines 28-37). *The Examiner notes that a free list is present in the system of McMahon. As shown in the table in Column 6 of McMahon, the dynamic memory allocator, in the form of these free lists, catalogs various free blocks of memory by size thereby indicating a boundary for each size that is maintained. Additional memory not fitting into the size constraints would be left as unused.*

As per dependent claim 9, McMahon teach, wherein a single type of data is stored in the block of memory (Column 3 lines 6-8). *The Examiner notes that the allocator of McMahon allocates memory to requesting programs of the operating system. The system designates each block that is allocated as used after allocation thereby eliminating reallocation of the block to a different program. Accordingly, the system of McMahon allows for a single type of data to be stored in the allocated block of memory.*

As per dependent claim 10, McMahon teach, wherein data from a fast query system is stored in the instances (Column 3 lines 6-8).

As per independent claim 11, McMahon teach,

- an application-level memory manager operable to allocate a block of memory to store data elements, (Column 5 lines 25-27)
- divide the block of memory into frames, and (Column 4 lines 37-40)
- divide each frame into instances; and application code operable to define data elements as having an instance type, and to associate the instance type with the instances for storage of the data elements in the instances.

(Column 5 lines 50-59).

- *The Examiner notes that the dynamic memory allocator takes the blocks (frames) from the whole memory and subsequently divides the blocks (frames) into portions (instances) for allocations. As the request for memory from the application, comes from the application, any allocated memory for that application will contain space available for that application to use. Accordingly, the allocation of memory for the application will contain data storage for the instances using the instances of the requesting application.*

As per dependent claim 12, McMahon teach, wherein the block of memory begins on a page boundary (Column 3 lines 28-37). *The Examiner notes that a free list is present in the system of McMahon. As shown in the table in Column 6 of McMahon, the dynamic memory allocator, in the form of these free lists, catalogs various free blocks of memory by size. Accordingly, when an allocation takes place by the dynamic memory allocator, for example of the 16-byte size, the memory allocated will begin on the 16-byte boundary.*

As per dependent claim 13, McMahon teach, wherein the size of the block of memory is determined by a coding parameter (Column 3 lines 6-8).

As per dependent claim 14, McMahon teach, wherein the block of memory includes a first page boundary (Column 3 lines 28-37). *The Examiner notes that a free list is present in the system of McMahon. As shown in the table in Column 6 of McMahon, the dynamic memory allocator, in the form of these free lists, catalogs*

various free blocks of memory by size thereby indicating a boundary for each size that is maintained.

As per dependent claim 15, McMahon teach, wherein a portion of the block of memory before the first page boundary is designated as unused (Column 6 lines 34-37). *The Examiner notes that as discussed supra and with respect to the instant citation, the allocator is able to divide the memory block and release the remainder of the block as free space. Accordingly, the allocator designates a portion of the block of memory as unused.*

As per dependent claim 16, McMahon teach, wherein a size of each frame is determined by a coding parameter (Column 3 lines 6-8).

As per dependent claim 17, McMahon teach, wherein a size of each frame is determined by the page size used by the operating system (Column 5 lines 8-21). *The Examiner notes that the allocator allocates virtual pages that originate from the operating system. Accordingly, the size of each frame is determined by the operating system's page size upon allocation.*

As per dependent claim 18, McMahon teach, wherein the block of memory includes a last page boundary and a portion of the block of memory after the last page boundary is designated as unused (Column 3 lines 28-37). *The Examiner notes that a free list is present in the system of McMahon. As shown in the table in Column 6 of McMahon, the dynamic memory allocator, in the form of these free lists, catalogs various free blocks of memory by size thereby indicating a boundary for each size that*

is maintained. Additional memory not fitting into the size constraints would be left as unused.

As per dependent claim 19, McMahon teach, wherein a single type of data is stored in the block of memory (Column 3 lines 6-8). *The Examiner notes that the allocator of McMahon allocates memory to requesting programs of the operating system. The system designates each block that is allocated as used after allocation thereby eliminating reallocation of the block to a different program. Accordingly, the system of McMahon allows for a single type of data to be stored in the allocated block of memory.*

As per dependent claim 20, McMahon teach, wherein the application code implements a fast query system (Column 3 lines 6-8).

As per independent claim 21, McMahon teach,

- associating data elements used by an application with an application-defined instance type; associating the application-determined instance type with an application-determined one of a plurality of blocks of memory allocated by an operating system, wherein the application-determined memory block is divided into frames that are further divided into instances; and *The Examiner incorporates herein by reference the rejections made supra with respect to claims 1 and 11.*
- populating the instances with the data elements (Column 3 lines 6-8).

As per dependent claim 22, McMahon teach, wherein associating the application-determined instance type with the application-determined block of memory comprises

associating a single application-determined instance type with the application-determined block of memory (Column 3 lines 6-8).

As per dependent claim 23, McMahon teach, removing the data elements; and returning the block of memory to the operating system (Column 62-65). *The Examiner notes that in the system of McMahon, the realloc() function call allocates a new block of memory and frees the original memory block. This process of allocating a new block teaches the instant limitation of removing the data elements. Once the block is freed, the dynamic memory allocator will find this block upon future searches.*

As per dependent claim 24, McMahon, returning the block of memory to a buffer; and determining after a predetermined period of time that the block of memory is no longer required by the application (Column 62-65).

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

1. U.S. Patent Application Publication 2003/0084266 Knippel et al teach a demand based memory block splitting method.
2. U.S. Patent Application Publication 2004/0073763 Dageville et al teach a dynamic and automatic memory management method.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Matthew Bradley whose telephone number is (571) 272-8575. The examiner can normally be reached on 6:30-3:00 M-F.

Art Unit: 2187

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Donald A. Sparks can be reached on (571) 272-4201. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DAS/mb



DONALD SPARKS
SUPERVISORY PATENT EXAMINER